An Hinf LPV design for sampling varying controllers : experimentation with a T inverted pendulum

David Robert, Olivier Sename, Daniel Simon

An H_∞ LPV design for sampling varying controllers : experimentation with a T inverted pendulum

David Robert ⋆, Olivier Sename⋆ and Daniel Simon◇

***Abstract*—This paper deals with the adaptation of a real-time controller's sampling period to account for the available computing resource variations. The design of such controllers requires a parameter-dependent discrete-time model of the plant, where the parameter is the sampling period. A polytopic approach for LPV (Linear Parameter Varying) systems is then developed to get an H_∞ sampling period dependent controller. A reduction of the polytope size is here performed which drastically reduces the conservatism of the approach and makes easier the controller implementation. Some experimental results on a T inverted pendulum are provided to show the efficiency of the approach.**

***Index Terms*—Digital control, linear parameter varying systems, H_∞ control, real experiments.**

I. Introduction

Some current trends in computer-controlled systems deal with the optimisation of computing resources consumption, in particular by increasing the flexibility of the system by on-line adaptation of the processor utilisation. In this way a feedback controller with a sampling period dependent PID controller is used in [1]. In [2], [3] a feedback scheduler based on a LQ optimisation of the control tasks periods is proposed. In [4] a processor load regulation is proposed and applied for real-time control of a robot arm. The design of a sampling period dependent RST controller was proposed in [5] for linear SISO systems. In [6], the case of LPV sampled-data systems is considered, where the sampling interval depends on the system parameters. The methodology to design output feedback controllers is based on the use of parameter dependent Lyapunov functions, which leads to an infinite number of inequalities to be solved. In [7] a "gridding" approach is used to design an observer-based state feedback controller with time-varying sampling period.

This paper provides a methodology for designing a sampling period dependent controller with performance adaptation, which can be used in the context of embedded control systems. The presented contribution enhances both previous papers [8], [9], using the linear robust control framework for Linear Parameter Varying (LPV) systems [10], where the sampling period is a parameter of the discrete-time model. In particular, the preliminary version of this paper has been presented in [9], from which we have improved some methodological developments. Indeed [9] considers a LPV approach where the polytope accounts for the parameters set $[h, h^2, \ldots, h^N]$. This means that the vertices of the polytope depend on the absolute values of the sampling period, while in this paper, we will use a less conservative formulation, considering the vector of parameters defined as the deviation of the sampling period h from its nominal value h_0. Also the whole design methodology is described here, including the way to get the polytopic model and, the discrete-time augmented plant from the polytopic model and from the weighting functions state space representations. This was not given in [9]. On the other hand, the reconstruction of the LPV controller, through the calculation of the polytopic coordinates, is explicitly given here in the general case, as a function of the deviation parameter, while, in [9], this part is not detailed at all and relies on the absolute value of the sampling period. We finally emphasize that these modifications have allowed to unify the notations along the paper, which was not the case in [9].

Note that three principal methods have been developed to deal with observation and control design for LPV systems. The polytopic approach uses the parameter values at each vertex of a polytope. Part of conservatism comes from the size of the polytope and in particular from the number of vertices of the polytope. Then some works tend to reduce the size of the polytope as in [11]. The gridding method [12] is a simple way to overcome some non linear or non affine terms in the inequalities. The main drawbacks of such an approach is the large number of LMIs to be solved brought by the discretization scheme of the parameter space and the loss of information between the gridding points. Finally the LFT approach [13] consists in pulling out the varying parameters, as done usually with uncertainties in robust control. It may lead to a exponential number of LMIs. Each method has in fact its own benefits and disadvantages.

In this paper we propose a parametrised discretization of the continuous time plant and of the weighting functions, leading to a discrete-time sampling period dependent augmented plant. In particular the plant discretization approximates the matrix exponential by a Taylor series of order N. Therefore we obtain a polytopic LPV model made of 2^N vertices, as presented in [8]. In this paper we exploit the dependency between the variables parameters, which are the successive powers of the sampling period, to reduce the number of controllers to be combined to $N+1$. The H_∞ control design method for polytopic models [10] is then used to get a sampling period dependent discrete-time controller. The reduction of the polytopic set drastically decreases both the complexity and the conservatism of the previous work and makes the solution easier to implement. This approach is then validated by experiments on real-time control of a T inverted pendulum.

The outline of this paper is as follows. Section II describes the reduced LPV discrete-time model. In section III the closed-

to appear In IEEE Transactions on Control Systems Technology

⋆ D.Robert and O. Sename are with GIPSA-lab (Control Systems Dpt., former LAG), UMR INPG-CNRS 5216, ENSE3-BP 46, 38402 Saint Martin d'Hères Cedex, France. Olivier.Sename@gipsa-lab.inpg.fr

◇ D. Simon is with INRIA Rhône-Alpes, Inovallée 655 avenue de l'Europe, Montbonnot, 38334 Saint-Ismier Cedex, France. Daniel.Simon@inrialpes.fr

loop objectives are stated and the augmented plant is built. Section IV presents the solution to the H_∞/LPV control design. Experiments on the "T" inverted pendulum are described in section V. Finally, the paper ends with some conclusions.

II. A SAMPLE DEPENDENT LPV DISCRETE-TIME MODEL

In this section the way to obtain a polytopic discrete-time model, the parameter of which being the sampling period, is detailed.

We consider a state space representation of continuous time plants as :

$$G : \left\{ \begin{array}{rcl} \dot{x} & = & Ax + Bu \\ y & = & Cx + Du \end{array} \right. \quad (1)$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ and $y \in \mathbb{R}^p$. The exact discretization of this system with a zero order hold at the sampling period h leads to the discrete-time LPV system (2)

$$G_d : \left\{ \begin{array}{rcl} x_{k+1} & = & A_d(h)\, x_k + B_d(h)\, u_k \\ y_k & = & C(h)\, x_k + D\, u_k \end{array} \right. \quad (2)$$

where

$$A_d(h) = e^{Ah} \qquad B_d(h) = \int_0^h e^{A\tau} d\tau B, \quad (3)$$

are usually computed as (see [14]):

$$\begin{pmatrix} A_d(h) & B_d(h) \\ 0 & I \end{pmatrix} = exp\left(\begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix} h \right) \quad (4)$$

with h ranging in $[h_{min}; h_{max}]$. However in (4) A_d and B_d are not affine on h.

A. Preliminary approach: Taylor expansion

Since h is assumed to belong to the interval $[h_{min},\ \ h_{max}]$ with $h_{min} > 0$, the sampling period is approximated around the nominal value h_0 of the sampling period, as:

$$h = h_0 + \delta \qquad \text{with} \qquad h_{min} - h_0 \leq \delta \leq h_{max} - h_0 \quad (5)$$

Then we can write:

$$\begin{pmatrix} A_d(h) & B_d(h) \\ 0 & I \end{pmatrix} = \begin{pmatrix} A_{h_0} & B_{h_0} \\ 0 & I \end{pmatrix} \begin{pmatrix} A_\delta & B_\delta \\ 0 & I \end{pmatrix} \quad (6)$$

where

$$\begin{pmatrix} A_{h_0} & B_{h_0} \\ 0 & I \end{pmatrix} := exp\left(\begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix} h_0 \right) \quad (7)$$

and

$$\begin{pmatrix} A_\delta & B_\delta \\ 0 & I \end{pmatrix} := exp\left(\begin{pmatrix} A & B \\ 0 & 0 \end{pmatrix} \delta \right) \quad (8)$$

In order to get a polytopic model, a Taylor series of order N is used to approximate the matrix exponential in (8), and allows to get:

$$A_\delta \approx I + \sum_{i=1}^{N} \frac{A^i}{i!}\, \delta^i \quad (9)$$

$$B_\delta \approx \sum_{i=1}^{N} \frac{A^{i-1}B}{i!}\, \delta^i \quad (10)$$

This leads to

$$A_d(h) = A_{h_0} A_\delta, \quad B_d(h) = B_{h_0} + A_{h_0} B_\delta \quad (11)$$

To evaluate the approximation error due to the Taylor approximation, the following criterion is used:

$$J_N = \max_{h_{min} < h < h_{max}} \| G_{d_e}(h) - G_d(h) \|_\infty \quad (12)$$

where G_{d_e} and G_d are the discrete-time models using the exact method (4) and the approximated one (9-10) respectively.

B. A first polytopic model

Let us define $H = [\delta, \delta^2, \ldots, \delta^N]$ the vector of parameters, that belongs to a convex polytope (hyper-polygon) $\mathcal{H}$ with 2^N vertices.

$$\mathcal{H} = \left\{ \sum_{i=1}^{2^N} \alpha_i(\delta)\omega_i : \alpha_i(\delta) \geq 0, \sum_{i=1}^{2^N} \alpha_i(\delta) = 1 \right\} \quad (13)$$

$$\{\delta, \delta^2, \ldots, \delta^N\},\ \ \delta^i \in \{\delta^i_{min}, \delta^i_{max}\} \quad (14)$$

Each vertex is defined by a vector $\omega_i = [\nu_{i_1}, \nu_{i_2}, \ldots, \nu_{i_N}]$ where ν_{i_j} can take the extremum values $\{\delta^j_{min},\ \delta^j_{max}\}$ with $\delta_{min} = h_{min} - h_0$ and $\delta_{max} = h_{max} - h_0$.

The matrices $A_d(\delta)$ and $B_d(\delta)$ are therefore affine in H and given by the polytopic forms:

$$A_d(H) = \sum_{i=1}^{2^N} \alpha_i(\delta) A_{d_i}, \quad B_d(H) = \sum_{i=1}^{2^N} \alpha_i(\delta) B_{d_i}$$

where the matrices at the vertices, i.e. A_{d_i} and B_{d_i}, are obtained by the calculation of $A_d(\delta)$ and $B_d(\delta)$ at each vertex of the polytope $\mathcal{H}$. The polytopic coordinates α_i which represent the position of a particular parameter vector $H(\delta)$ in the polytope $\mathcal{H}$ are given solving :

$$H(\delta) = \sum_{i=1}^{2^N} \alpha_i(\delta)\omega_i \quad , \quad \alpha_i(\delta) \geq 0 \quad , \quad \sum_{i=1}^{2^N} \alpha_i(\delta) = 1 \quad (15)$$

As an illustration, figure 1 shows this transformation for $N = 2$ with $H_1 = [\delta_{min}, \delta^2_{min}]$, $H_2 = [\delta_{max}, \delta^2_{min}]$, $H_3 = [\delta_{min}, \delta^2_{max}]$, $H_4 = [\delta_{max}, \delta^2_{max}]$.

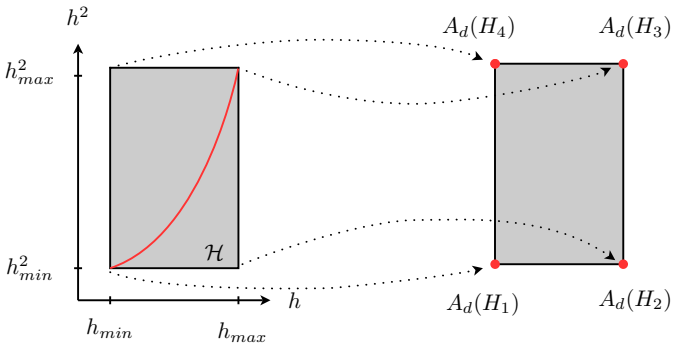


Fig. 1. Example of polytope for $A_d(\delta)$ with $\delta_{min} = 0$

This leads to the plant polytopic model (16) where G_{d_i} are $G_d(H)$ evaluated at the vertices ω_i.

$$G_d(H) = \sum_{i=1}^{2^N} \alpha_i(\delta) G_{d_i} \quad \text{and} \quad H = \sum_{i=1}^{2^N} \alpha_i(\delta)\omega_i \quad (16)$$

As the gain-scheduled controller will be a convex combination of 2^N "vertex" controllers, the choice of the series order N gives a trade-off between the approximation accuracy and the controller complexity. Indeed one should notice that:

- The raw approach does not take into account the dependence between $\delta, \delta^2, \ldots, \delta^N$. Indeed, as shown in figure 1, the set of parameters $\{[\delta, \delta^2], 0 \leq \delta \leq \delta_{max}\}$, represented by the parabolic curve, is included in the large polytopic box with 4 vertices. This will of course induce some conservatism in the control design.
- Moreover, when a the order of the Taylor approximation increases, we will see that the number of LMIs to be solved, which is $2\times 2^N+1$ will grow exponentially which can lead to unfeasible optimisation problems.
- Finally the implementation of the controller is also directly linked to the number of vertices of the polytope.

To reduce the complexity (and the conservatism of the corresponding control design as well), a reduction of the polytope is proposed below.

C. Reduction of the polytope

It is here proposed to reduce the size of the polytope using the dependency between the successive powers of the parameter δ. This reduction only stands for $\delta_{min} = 0$, which means that $h_0 = h_{min}$ is the smallest sampling period, i.e. related with a slack constraint on computing resource. For control purpose this choice is quite logical as the nominal behaviour corresponds to the minimal sampling period in normal situations. This period would increase only when computing resources will be limited.

The way to reduce the size of the polytopic set can be seen on the example in figure 1, where the parabolic parameters locus is enclosed in the triangle defined by $\{0, 0\}$, $\{\delta_{max}, 0\}$ and $\{\delta_{max}, \delta^2_{max}\}$. Therefore it is not necessary to consider the vertex $\{0, \delta^2_{max}\}$ to build a polytope encompassing the parameters locus. To develop and extend this method to a polytope of size N, let us write:

$$h = h_{min} + \delta, \quad 0 \leq \delta \leq \delta_{max}, \quad \delta_{max} = h_{max} - h_{min}, \tag{17}$$

Then the inequality below is always satisfied:

$$\delta \ \delta^n \leq \frac{\delta^{n+1}_{max}}{\delta^n_{max}} \ \delta^n \quad \text{i.e.} \quad \delta^{n+1} \leq \delta_{max} \ \delta^n \tag{18}$$

Then it is proposed to delete the vertices which do not satisfy the above inequality. As the vertices H_i of $\mathcal{H}$ are given by a vector $(\nu_1, \nu_2, \ldots, \nu_N)$ where $\nu_i = 0$ or δ^i_{max} according to the considered vertex, the inequality to be satisfied is:

$$\nu_{n+1} \leq \delta_{max} \ \nu_n \tag{19}$$

This leads to the following set of admissible vertices: $(0, 0, 0, \ldots, 0)$, $(\delta_{max}, 0, 0, \ldots, 0)$, $(\delta_{max}, \delta^2_{max}, 0, \ldots, 0)$, $\ldots$ $(\delta_{max}, \delta^2_{max}, \delta^3_{max}, \ldots, \delta^N_{max})$. This method leads to a set of $N + 1$ vertices instead of 2^N. Note that these vertices are linearly independent and make a simplex, which is itself basically a polytope [15] of minimal dimension considering the parameters space of dimension N. When $N = 2$ (and for $0 < \delta < \delta_{max}$) the square is downsized to a triangle and a pyramid is the reduction of a hexahedron (for $N = 3$).

III. Formulation of the H_∞/LPV control problem

In this section we first present the formulation of the H_∞ control problem using weighting function depending on the sampling period. Indeed the provided methodology will allow for performance adaptation according to the computing resources availability. The H_∞ framework is based on the general control configuration of figure 2, where W_i and W_o are some weighting functions representing the specification of the desired closed-loop performances. The objective is here to find a controller K such internal stability is achieved and $\|\tilde{z}\|_2 < \gamma \|\tilde{w}\|_2$, where γ represents the H_∞ attenuation level.

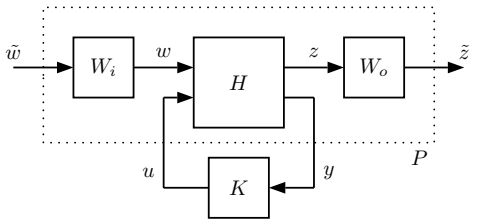


Fig. 2. Focused interconnection

Classical control design assumes constant performance objectives and produces a controller with an unique sampling period. The sampling period is chosen according to the controller bandwidth, the noise sensibility and the availability of computation resources. When the sampling period varies the usable controller bandwidth also varies and the closed-loop objectives should logically be adapted. Therefore the bandwidth of the weighting functions is chosen to be dependent on the sampling period to allow for performance adaptation. Indeed trying to achieve high level performances for a range of sampling period might lead to instability and/or robustness problems.

A. Towards discrete-time adaptive weighting functions

The methodology is as follows. First W_i and W_o are split into two parts :

- a constant part with constant poles and zeros. This allows, for instance, to compensate for oscillations or flexible modes which are, by definition, independent of the sampling period.
- the variable part contains poles and zeros whose pulsations are expressed as an affine function of the sampling frequency $f = 1/h$. This allows for an adaptation of the bandwidth of the weighting functions, and hence for an adaptation of the closed-loop performance w.r.t. the available computing power. These poles and zeros are here constrained to be *real* by the discretization step.

First of all the constant parts of the weighting functions are merged with the continuous-time plant model. Then a discrete-time augmented system is developed as presented above.

The variable part $V(s)$ of a weighting function is the discretized according to the following methodology:

1) factorise $V(s)$ as a product of first order systems. We here chose poles and zeros depending linearly of the

sampling frequency $f = 1/h$, as:

$$V(s) = \beta \prod_i \frac{s - b_i f}{s - a_i f} = \beta \prod_i V_i(s) \tag{20}$$

with $a_i, b_i \in \mathbb{R}$

2) Give $V_i(s)$ as the state space observable canonical form

$$V_i(s) : \begin{cases} \dot{x}_i = a_i f \ x_i + f(a_i - b_i) \ u_i \\ y_i = x_i + u_i \end{cases} \tag{21}$$

3) Form the series interconnection of all $V_i(s)$. This leads to:

$$V(s) : \begin{cases} \dot{x}_v &= A_v f \ x_v + B_v f \ u_v \\ y_v &= C_v \ x_v + D_v \ u_v \end{cases} \tag{22}$$

4) Get the discrete-time state space representation of $V(s)$. Thanks to the affine dependence in f in (22) the discrete-time model of the variable part becomes independent of h since:

$$\begin{cases} A_{v_d} = e^{A_v f \, h} = e^{A_v} \\ B_{v_d} = (A_v f)^{-1}(A_{v_d} - I)B_v f = (A_v)^{-1}(A_{v_d} - I)B_v \\ C_{v_d} = C_v \text{ and } D_{v_d} = D_v \end{cases} \tag{23}$$

Remark 1: The simplification between f and h in (23) *simplifies the discretization step. This is why the plant and the weighting functions are separately discretized, and the augmented plant is obtained in discrete time afterwards by interconnection. This is also a consequence of the use of the observable canonical form.*

B. The discrete-time augmented plant

Let first consider the following continuous-time model where the constant part of the weighting function W_i and W_o has been connected to the plant model:

$$P : \begin{cases} \dot{x}(t) = Ax(t) + B_w w(t) + B_u u(t) \\ z(t) = C_z x(t) + D_{zw} w(t) + D_{zu} u(t) \\ y(t) = C_y x(t) + D_{yw} w(t) + D_{yu} u(t) \end{cases} \tag{24}$$

where $x \in R^n$ is the state, $w \in R^{m_w}$ represents the exogenous inputs, $u \in R^{m_u}$ the control inputs, $z \in R^{p_z}$ the controlled output and $y \in R^{p_y}$ the measurement vector.

A discrete-time representation of the above system is first obtained thanks to the previous methodology. For simplicity we will note, according to the representation (1):

$$B = \begin{pmatrix} B_w & B_u \end{pmatrix}, \quad C = \begin{pmatrix} C_z \\ C_y \end{pmatrix}, \quad D = \begin{pmatrix} D_{zw} & D_{zu} \\ D_{yw} & D_{yu} \end{pmatrix} \tag{25}$$

Using the Taylor approximation at order N leads to a polytope $\mathcal{H}$. This polytope has r vertices (where r equals 2^N for the basic case and $N+1$ for the reduced one). Each of the r vertices is described by a vector ω_i of the form $(\delta_1, \delta_2, \ldots, \delta_r)$ where $\delta_i = \delta^i_{min}$ or δ^i_{max}.

The LPV polytopic discrete-time model is then given by:

$$\mathcal{P}(H) : \begin{cases} x_{k+1} = \mathcal{A}(H)x_k + \mathcal{B}_w(H)w_k + \mathcal{B}_u(H)u_k \\ z_k = C_z x_k + D_{zw} w_k + D_{zu} u_k \\ y_k = C_y x_k + D_{yw} w_k + D_{yu} u_k \end{cases} \tag{26}$$

$$\begin{aligned} H &= \begin{pmatrix} \delta & \delta^2 & \ldots & \delta^N \end{pmatrix} = \sum_{i=1}^{r} \alpha_i \omega_i, \\ \mathcal{A}(H) &= \sum_{i=1}^{r} \alpha_i \mathcal{A}_i, \quad \sum_{i=1}^{r} \alpha_i = 1, \ \alpha_i \geq 0 \end{aligned} \tag{27}$$

where, according to the representation (2)

$$A_d = \mathcal{A}, \quad B_d = \begin{pmatrix} \mathcal{B}_w & \mathcal{B}_u \end{pmatrix}, \quad C_d = C, \quad D_d = D \tag{28}$$

Now, the variable part of the weighting functions W_i and W_o are expressed as previously presented, which leads to both discrete-time representations (29) and(30) where the size of the state vector depend on the weighting function:

$$\mathcal{W}_I : \begin{cases} x_{I_{k+1}} = A_I x_{I_k} + B_I \tilde{w} \\ w = C_I x_{I_k} + D_I \tilde{w} \end{cases} \tag{29}$$

$$\mathcal{W}_O : \begin{cases} x_{O_{k+1}} = A_O x_{O_k} + B_O z \\ \tilde{z} = C_O x_{O_k} + D_O z \end{cases} \tag{30}$$

The augmented system $\mathcal{P}'(H)$ is obtained by the interconnection of $\mathcal{P}(H)$, $\mathcal{W}_I$ and $\mathcal{W}_O$. Therefore we obtain the following LPV polytopic discrete-time system of state vector $x'_k = (x_k \ x_{I_k} \ x_{O_k})^T$:

$$\mathcal{P}'(H) : \begin{cases} x'_{k+1} = \mathcal{A}'(H)x'_k + \mathcal{B}'_w(H)\tilde{w} + \mathcal{B}'_u(H)u \\ \tilde{z} = \mathcal{C}'_z x'_k + \mathcal{D}'_{zw}\tilde{w} + \mathcal{D}'_{zu}u \\ y = \mathcal{C}'_y x'_k + \mathcal{D}'_{yw}\tilde{w} + \mathcal{D}'_{yu}u \end{cases} \tag{31}$$

$$\mathcal{A}'(H) = \begin{pmatrix} \mathcal{A}(H) & \mathcal{B}_w(H)C_I & 0 \\ 0 & A_I & 0 \\ B_O C_z & B_O D_{zw} C_I & A_O \end{pmatrix}$$

$$\mathcal{B}'_w(H) = \begin{pmatrix} \mathcal{B}_w(H)D_I \\ B_I \\ B_O D_{zw} D_I \end{pmatrix}, \ \mathcal{B}'_u(H) = \begin{pmatrix} \mathcal{B}_u(H) \\ 0 \\ B_O D_{zu} \end{pmatrix}$$

$\mathcal{C}'_z = \begin{pmatrix} D_O C_z & D_O D_{zw} C_I & C_O \end{pmatrix}$, $\mathcal{D}'_{zw} = \begin{pmatrix} D_O D_{zw} D_I \end{pmatrix}$, $\mathcal{D}'_{zu} = \begin{pmatrix} D_O D_{zu} \end{pmatrix}$, $\mathcal{C}'_y = \begin{pmatrix} C_y & D_{yw} C_I & 0 \end{pmatrix}$, $\mathcal{D}'_{yw} = \begin{pmatrix} D_{yw} D_I \end{pmatrix}$, $\mathcal{D}'_{yu} = \begin{pmatrix} D_{yu} \end{pmatrix}$.

IV. Solution to the H_∞ control problem for LPV systems

We consider here LPV systems of the form (31). The method in [10] requires the following assumptions:

(A1) $\mathcal{D}'_{yu}(H) = 0$

(A2) $\mathcal{B}'_u(H), \mathcal{C}'_y, \mathcal{D}'_{zu}, \mathcal{D}'_{yw}$ are parameter- independent

(A3) the pairs $(\mathcal{A}'(H), \mathcal{B}'_u(H))$ and $(\mathcal{A}'(H), \mathcal{C}'_y)$ are quadratically stabilisable and detectable over H respectively,

Remark 2: In (31) *assumption (A2) is not satisfied due to the $B_u(H)$ term in $\mathcal{B}'_u(H)$. To avoid this, a strictly proper filter is added on the control input, as explained in [10]. It is a numerical artifact (which of course increases the number of state variables $n_e > n_x$), therefore its bandwidth should be chosen high enough to be negligible regarding the plant and objective bandwidths.*

A. Problem solution

Proposition 1: Following [10] , under the previous assumptions there exists a gain-scheduled controller

$$\left\{ \begin{array}{rcl} x_{K_{k+1}} & = & A_K(H)x_{K_k} + B_K(H)y_k \\ u_k & = & C_K(H)x_{K_k} + D_K(H)y_k \end{array} \right. \quad (32)$$

where $x_K \in \mathbb{R}^{n_e}$, which ensures, over all parameter trajectories, that :

- the closed-loop system is internally quadratic stable;
- the $\mathscr{L}_2$-induced norm of the operator mapping w into z is bounded by γ, i.e. $\|z\|_2 < \gamma\|w\|_2$

if and only if there exist γ and two symmetric matrices (R, S) satisfying $2r+1$ LMIs (which are computed off-line) :

$$\left(\begin{array}{c|c} N_R & 0 \\ \hline 0 & I \end{array}\right)^T \mathcal{L}_1^i \left(\begin{array}{c|c} N_R & 0 \\ \hline 0 & I \end{array}\right) < 0, i = 1 \dots r \quad (33)$$

$$\left(\begin{array}{c|c} N_S & 0 \\ \hline 0 & I \end{array}\right)^T \mathcal{L}_2^i \left(\begin{array}{c|c} N_S & 0 \\ \hline 0 & I \end{array}\right) < 0, i = 1 \dots r \quad (34)$$

$$\left(\begin{array}{cc} R & I \\ I & S \end{array}\right) \geq 0 \quad (35)$$

where

$$\mathcal{L}_1^i = \left(\begin{array}{cc|c} \mathcal{A'}_i R \mathcal{A'}_i^T - R & \mathcal{A'}_i R \mathcal{C'}_{zi}^T & \mathcal{B'}_{wi} \\ \mathcal{C'}_{zi} R \mathcal{A'}_i^T & -\gamma I + \mathcal{C'}_{zi} R \mathcal{C'}_{zi}^T & \mathcal{D'}_{zwi} \\ \hline \mathcal{B'}_{wi}^T & \mathcal{D'}_{zwi}^T & -\gamma I \end{array}\right)$$

$$\mathcal{L}_2^i = \left(\begin{array}{cc|c} \mathcal{A'}_i^T S \mathcal{A'}_i - S & \mathcal{A'}_i^T S \mathcal{B'}_{wi} & \mathcal{C'}_{zi}^T \\ \mathcal{B'}_{wi}^T S \mathcal{A'}_i & -\gamma I + \mathcal{B'}_{wi}^T S \mathcal{B'}_{wi} & \mathcal{D'}_{zwi}^T \\ \hline \mathcal{C'}_{zi} & \mathcal{D'}_{zwi} & -\gamma I \end{array}\right)$$

where $\mathcal{A'}_i$, $\mathcal{B'}_{wi}$, $\mathcal{C'}_{zi}$, $\mathcal{D'}_{zwi}$ are $\mathcal{A'}(H)$, $\mathcal{B'}_w(H)$, $\mathcal{C'}_z(H)$, $\mathcal{D'}_{zw}(H)$ evaluated at the i^{th} vertex of the parameter polytope. N_S and N_R denote bases of null spaces of $(\bar{B}_2^T, \bar{D}_{12}^T)$ and $(\bar{C}_2, \bar{D}_{21})$ respectively.

B. Controller reconstruction

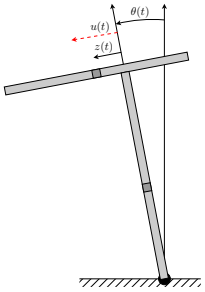
Once R, S and γ are obtained, the controllers are reconstructed at each vertex of the parameter polytope as shown in [10]. The gain-scheduled controller $K(H)$ is then the convex combination of these controllers

$$K(H) : \left(\begin{array}{cc} A_K(H) & B_K(H) \\ C_K(H) & D_K(H) \end{array}\right) = \sum_{i=1}^{r} \alpha_i(\delta) \left(\begin{array}{cc} A_{K_i} & B_{K_i} \\ C_{K_i} & D_{K_i} \end{array}\right) \quad (36)$$

$$\text{with } \alpha_i(\delta) \text{ such that } H = \sum_{i=1}^{r} \alpha_i(\delta)\omega_i \quad (37)$$

Note that on-line scheduling of the controller needs the computation of $\alpha_i(\delta)$ knowing h. For the full polytope case the polytopic coordinates are solutions of the following under-constrained system:

$$\left\{ \begin{array}{l} \sum_{i=1}^{2^N} \alpha_i(\delta)\omega_i = H = [\delta, \delta^2, ..., \delta^N] \\ \sum_{i=1}^{2^N} \alpha_i(\delta) = 1, \ \alpha_i(\delta) \geq 0 \end{array} \right. \quad (38)$$



m_1	0.217 kg	horizontal sliding rod mass
m_2	1.795 kg	vertical rod mass
l_0	0.33	vertical rod length
l_c	-0.032 m	vertical rod position of the centre of gravity
g	9.81 m.s^{-2}	gravity acceleration
$\bar{J}$	0.061 Nm^2	Nominal inertia
f_{v_z}	0.1 kg.s^{-1}	viscous friction

Fig. 3. Coordinates of the T pendulum

which can be solved using an algorithm of the LMI toolbox [10]. When the polytope is reduced to a simplex (using inequality (19)) the polytopic coordinates are given solving a simpler system :

$$\left\{ \begin{array}{l} \sum_{i=1}^{N+1} \alpha_i(\delta)\omega_i = H = [\delta, \delta^2, ..., \delta^N] \\ \sum_{i=1}^{N+1} \alpha_i(\delta) = 1, \ \alpha_i(\delta) \geq 0 \end{array} \right. \quad (39)$$

for which explicit solutions are easily recursively computed:

$$\left\{ \begin{array}{l} \alpha_1 = \frac{\delta_{max} - \delta}{\delta_{max} - \delta_{min}} \\ \alpha_n = \frac{\delta_{max}^n - \delta^n}{\delta_{max}^n - \delta_{min}^n} - \sum_1^{n-1} \alpha_i \quad , \quad n = [2, ..., N] \\ \alpha_{N+1} = 1 - \sum_1^N \alpha_i \end{array} \right. \quad (40)$$

This leads, for the case $N = 2$ and $\delta_{min} = 0$ of the next section to the simple explicit solutions:

$$\alpha_1 = \frac{\delta_{max} - \delta}{\delta_{max}}, \alpha_2 = \frac{\delta_{max}^2 - \delta^2}{\delta_{max}^2} - \alpha_1, \alpha_3 = 1 - (\alpha_1 + \alpha_2)$$

V. Control of the T inverted pendulum

This section is devoted to an experimental validation of the approach using a "T" inverted pendulum of Educational Control Products[1], available at GIPSA-lab. These experiments will emphasise the effectiveness of the proposed design method. The pendulum is made of two rods. A vertical one which rotates around the pivot axle, and an horizontal sliding balance one. Two optional masses allow to modify the plant's dynamical behaviour. The control actuator (DC motor) delivers a force u to the horizontal sliding rod, through a drive gear-rack. The θ angle, positive in the trigonometric sense, is measured by the rod angle sensor. The position z of the horizontal rod is measured by a sensor located at the motor axle. The DC motor is torque controlled using a local current feedback loop (assumed to be a simple gain due to its fast dynamics). The dynamical behaviour of the sensors is also neglected.

A. Modelling

A mechanical model of the pendulum is presented below, which takes into account the viscous friction (but not the Coulomb friction). This model has been developed in [16].

$$\begin{pmatrix} m_1 & m_1 l_0 \\ m_1 l_0 & \bar{J} \end{pmatrix} \begin{pmatrix} \ddot{z} \\ \ddot{\theta} \end{pmatrix} + \begin{pmatrix} -f_{v_z} & -m_1 z\dot{\theta} \\ 2m_1 z\dot{\theta} & 0 \end{pmatrix} \begin{pmatrix} \dot{z} \\ \dot{\theta} \end{pmatrix} + \begin{pmatrix} -m_1 \sin\theta \\ -(m_1 l_0 + m_2 l_c)\sin\theta - m_1 z \cos\theta \end{pmatrix} g = \begin{pmatrix} u \\ 0 \end{pmatrix} \quad (41)$$

[1] http://www.ecpsystems.com/controls_pendulum.htm

Choosing the state vector as $x = [z, \dot{z}, \theta, \dot{\theta}]$, we get the following non linear state space representation:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -l_0 \dot{x}_4 + x_1 x_4^2 + g \sin x_3 - \dfrac{f_{v_z}}{m_1} x_2 + \dfrac{u}{m_1} \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \dfrac{1}{J_0(x_1) - m_1 l_0^2} \left(+g(m_1 x_1 \cos x_3 + m_2 l_c \sin x_3)\right. \\ \qquad \left. -m_1(l_0 x_4 + 2x_2) x_1 x_4 - l_0 u\right) \end{cases} \tag{42}$$

with $J_0(x_1) = \bar{J} + m_1 x_1^2$. The steady-state linearisation around $x = [0, 0, 0, 0]$ gives the linear state space representation $\dot{x}(t) = Ax(t) + Bu(t)$, $y(t) = Cx(t)$ with

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ \frac{-l_0 g m_1}{\bar{J} - m_1 l_0^2} & -\frac{f_{v_z}}{m_1} & \frac{-l_0 g m_2 l_c}{\bar{J} - m_1 l_0^2} + g & 0 \\ 0 & 0 & 0 & 1 \\ \frac{g m_1}{\bar{J} - m_1 l_0} & 0 & \frac{g m_2 l_c}{\bar{J} - m_1 l_0^2} & 0 \end{pmatrix},$$

$$B = \begin{pmatrix} 0 \\ \frac{l_0^2}{\bar{J} - m_1 l_0^2} + \frac{1}{m_1} \\ 0 \\ \frac{-l_0}{\bar{J} - m_1 l_0^2} \end{pmatrix}, C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

B. Performance specification

As such a T pendulum system is difficult to be controlled, our main objective is here to get a closed-loop stable system, to emphasize the practical feasibility of the proposed methodology for real-time control. From previous experiments with this plant the sampling period interval is set to $[1, 3]$ ms.

The chosen performance objectives are represented in figure 4, where the tracking error and the control input are weighted (as usual in the H_∞ methodology).

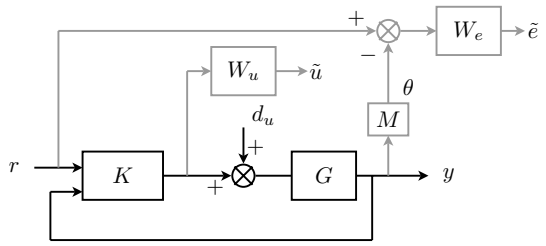


Fig. 4. General control configuration

This corresponds to the mixed sensitivity problem given in (43).

$$\left\| \begin{matrix} W_e(I - MS_yGK_1) & W_eMS_yG \\ W_uS_uK_1 & W_uT_u \end{matrix} \right\|_\infty \leq \gamma \tag{43}$$

with

$$\begin{aligned} K &= \begin{bmatrix} K_1 & K_2 \end{bmatrix} & M &= \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} \\ S_u &= (I - K_2 G)^{-1} & S_y &= (I - GK_2)^{-1} \\ T_u &= -K_2 G (I - K_2 G)^{-1} \end{aligned} \tag{44}$$

The performance objectives are represented by the usual weighting functions:

$$W_e(p, f) = \frac{p\, M_S + \omega_S(f)}{p + \omega_S(f)\, \epsilon_S} \qquad W_u(p, f) = \frac{1}{M_U} \tag{45}$$

where $\omega_S(f) = h_{min}\ \omega_{S_{max}}\ f$, $f = 1/h$, $\omega_{S_{max}}$ = 1,5 rad/s, M_S = 2, ϵ_S = 0.01 and M_U = 5.

Notice that only W_e depends on the sampling frequency to account for performance adaptation.

C. Polytopic discrete-time model

We follow here the methodology proposed in section II. The approximation is done around the nominal period $h_o = 1ms$, for $h \in [1, 3]$ ms, i.e. $\delta_h \in [0, 2]$ ms.

On figure 5 the criterion (12) is evaluated for different sampling periods ($h \in [1, 3]ms$) and different orders of the Taylor expansions ($k \in [1, 5]$). It shows that this error may be large only if the order 1 is used. Also, when the Taylor expansion orders equals 2 (resp. 4), the approximation error is less than $-40dB$ (resp. $-140dB$).

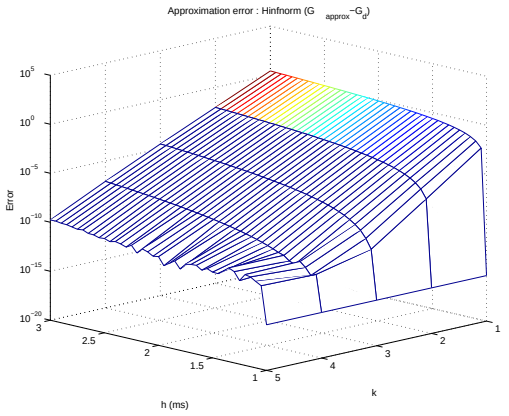


Fig. 5. Approximation error w.r.t h and Taylor order k

D. LPV/H_∞ design

The first step is the discretization of the weighting functions. The augmented system is got, using a preliminary first-order filtering of the control input, to satisfy the design assumptions. The augmented system is of order 6.

Applying the design method developed in section IV leads to the following results, combining the Taylor expansion order and the polytope reduction:

	Polytope	Nb vertices	γ_{opt}
Taylor order N=2	full	4	1.1304
Taylor order N=2	reduced	3	1.1299
Taylor order N=4	full	16	1.1313
Taylor order N=4	reduced	5	1.1303

This table emphasises that both design of orders 2 and 4 are reliable. For implementation reasons (simplicity and computational complexity) we have chosen the case of the reduced polytope using a Taylor expansion of order 2.

The corresponding sensitivity functions of the chosen design are shown in figure 6. Using $S_e = e/r$ the steady-state tracking

error is less than $-46dB$, with a varying bandwidth from 0.4 to 1.2 rad/s, i.e the ratio 3, specified according to the interval of sampling period, is satisfied. The peak value of $S_u K_1$ varies from 1.2 to $10.8dB$, which is reasonable for the control gain. Finally the function MS_yGd_u is very low so that the effect of input disturbance d_u on the tracking error will be greatly attenuated.

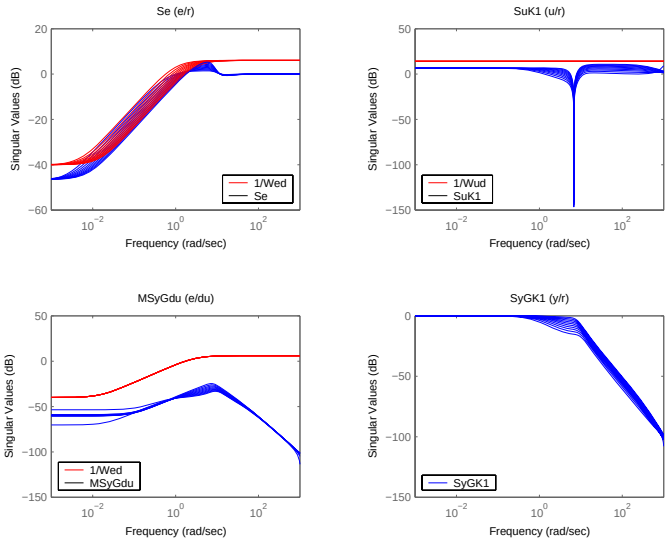


Fig. 6. Sensitivity functions

E. Simulation results

In this section, the application of the proposed sampling variable controller when the sampling period varies on-line between 1 and 3 msec. is provided.

Two cases are presented. First in figure 7 the sampling period variation is continuous and follows a sinusoidal signal of frequency $0.15rad/s$. Then in figure 8 some step changes of the sampling period are done.

These results show that, as expected from the performance specification, the settling time of the closed-loop system varies accordingly with the sampling period. When the period is large (i.e at $t = 10sec$) the pendulum is slower, while when the period is small (i.e at $t = 30sec$ in Fig. 7) the pendulum response is faster. Moreover, thanks to the LPV approach, the variations (sinusoidal or step changes) of the sampling period do not lead to abrupt transient of the pendulum behaviour. This is a great benefit from the LPV approach which ensures the stability for arbitrarily fast variations of the parameter in their allowed range (this is due to the use of a single Lyapunov function in the design [10]). The same assessment can be done for the control input.

The LPV scheme allows here to guarantee the closed-loop quadratic stability, to have a bounded $\mathscr{L}_2$-induced norm for all variation of the sampling period and to have a predictable closed-loop behaviour.

F. Experiments

The scenarii of the previous section (simulation results) are now implemented for the real plant. The plant is controlled

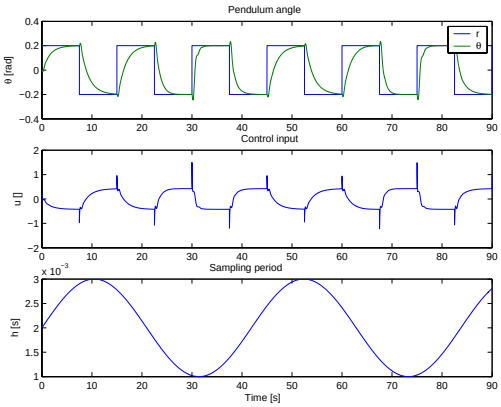


Fig. 7. Motion of the T pendulum under a sinusoidal sampling period

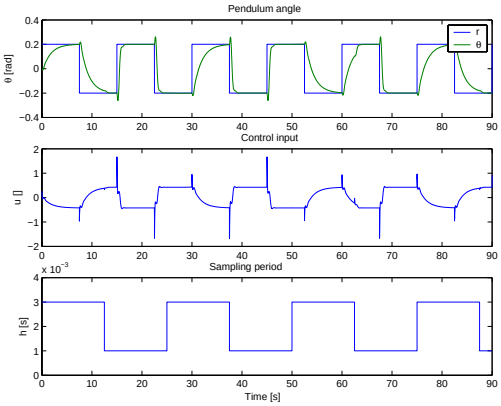


Fig. 8. Motion of the T pendulum under a square sampling period

through Matlab/Simulink using the Real-time Workshop and xPC Target.

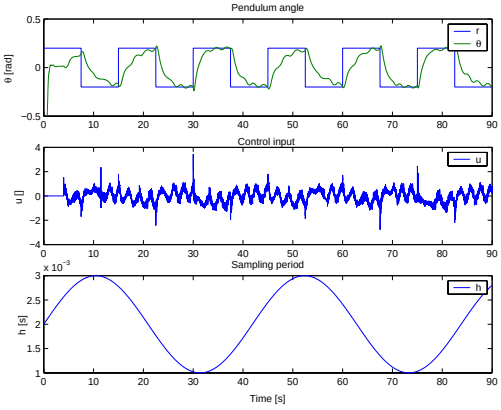


Fig. 9. Experimental motion of the T pendulum under a sinusoidal sampling period

The results are given in figures 9 and 10. As in the previous section, the settling time is maximal when the sampling period is maximal, and conversely. In the same way, there is no abrupt changes in the control input (even when the sampling period abruptly varies from 1 to 3 ms as in figure 10).

The noise on the control input is a consequence of the combination of dry friction and elasticity in the pendulum

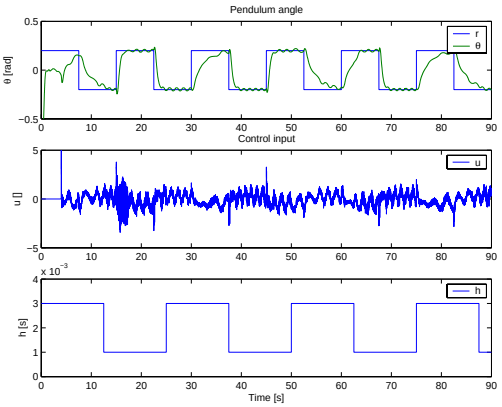


Fig. 10. Experimental motion of the T pendulum under a square sampling period

actuation. Note that incidentally the rather large value of the $S_u K_1$ sensitivity at high frequencies here induces a permanent dithering of the control signal: our particular benchmark benefits from the control noise to reduce the friction disturbances effect, a problem which is known to be difficult to take into account ([17]).

Finally we get similar results in simulation and experimental tests which shows the inherent robustness property of the H_∞ design. These results emphasize the great advantage and flexibility of the method when the available computing resources may vary, and when sampling interval variations are used to handle computing flexibility as in [4].

VI. Conclusion

In this paper, an LPV approach is proposed to design a discrete-time linear controller with a varying sampling period and varying performances. A way to reduce the polytope from 2^N to $N+1$ vertices (where N is the Taylor order expansion) is provided, which drastically reduces both the conservatism and the complexity of the resulting sampling dependent controller and makes the solution easier to implement. Further developments may concern the reduction of the conservatism which is due to to the use a constant Lyapunov function approach, which is known to produce a sub-optimal controller. Others approaches as in [12], [13], [18] could be employed.

The whole methodology has been implemented for the case of a "T" inverted pendulum, where experimental results have been provided. These results emphasise the real effectiveness of the LPV approach as well as its interest in the context of adaptation to varying processor or network load where a bank of switching controllers would need too much resources. Using a single controller synthesis, the stability and performance property of the closed-loop system are guaranteed whatever the speed of variations of the sampling period are. In addition we also observed an interesting robustness of this controller w.r.t. sampling inaccuracies, e.g. which could be induced by preemptions in a multi-tasking operating systems.

Note that the controller parameters depend on the sampling intervals, defined as the intervals between measurements incoming instants at the input of the controller's computation. However the use of this parameter is open : it may be given by a feedback scheduler (as usually in our research framework), which in this case gives the current sampling interval. It can be also a network-induced latency measured by the controller local clock (and is still a current value for the sampling interval).

As shown in preliminary studies ([4], [19]), these properties are of prime interest in the design of more complex systems combining several such controllers under supervision of a feedback-scheduler : the control intervals can be varied arbitrarily fast by an outer scheduling loop under a QoS objective with no risk of jeopardising the plants stability.

References

[1] A. Cervin and J. Eker, "Feedback scheduling of control tasks," in *Proceedings of the 39th IEEE Conference on Decision and Control*, Sydney, Australia, Dec. 2000.

[2] A. Cervin, J. Eker, B. Bernhardsson, and K. Årzén, "Feedback-feedforward scheduling of control tasks," *Real-Time Systems*, vol. 23, no. 1–2, pp. 25–53, Jul. 2002.

[3] J. Eker, P. Hagander, and K. Årzén, "A feedback scheduler for real-time controller tasks," *Control Engineering Practice*, vol. 8, no. 12, pp. 1369–1378, Dec. 2000.

[4] D. Simon, D. Robert, and O. Sename, "Robust control / scheduling co-design: application to robot control," in *Proceedings of the 11th IEEE Real-Time and Embedded Technology and Applications Symposium*, San Francisco, USA, March 2005.

[5] D. Robert, O. Sename, and D. Simon, "Sampling period dependent RST controller used in control/scheduling co-design," in *Proceedings of the 16th IFAC World Congress*, Prague, Czech Republic, Jul. 2005.

[6] K. Tan, K.-M. Grigoriadis, and F. Wu, "Output-feedback control of LPV sampled-data systems," *Int. Journal of Control*, vol. 75, no. 4, pp. 252–264, 2002.

[7] A. Sala, "Computer control under time-varying sampling period: An LMI gridding approach," *Automatica*, vol. 41, no. 12, pp. 2077–2082, 2005.

[8] D. Robert, O. Sename, and D. Simon, "Synthesis of a sampling period dependent controller using LPV approach," in *5th IFAC Symposium on Robust Control Design ROCOND'06*, Toulouse, France, july 2006.

[9] ——, "A reduced polytopic LPV synthesis for a sampling varying controller : experimentation with a t inverted pendulum," in *European Control Conference ECC'07*, Kos, Greece, July 2007.

[10] P. Apkarian, P. Gahinet, and G. Becker, "Self-scheduled H_∞ control of linear parameter-varying systems: A design example," *Automatica*, vol. 31, no. 9, pp. 1251–1262, 1995.

[11] J.-M. Biannic, "Commande robuste des systèmes à paramètres variables," Ph.D. dissertation, ENSAE, Toulouse, France, 1996.

[12] P. Apkarian and R. Adams, "Advanced gain-scheduling techniques for uncertain systems," *IEEE Transactions on Automatic Control*, vol. 6, pp. 21–32, 1998.

[13] A. Packard, "Gain scheduling via linear fractional transformations," *Systems and Control Letters*, vol. 22, pp. 79–92, 1994.

[14] K. J. Åström and B. Wittenmark, *Computer-Controlled Systems*, 3rd ed., ser. Information and systems sciences series. New Jersey: Prentice Hall, 1997.

[15] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[16] O.-R. Natale, O.Sename, and C. C. de Wit, "Inverted pendulum stabilization through the ethernet network, performance analysis," in *IEEE American Control Conference*, Boston, USA, June 30-July 2 2004.

[17] H. Olsson, K.-J. Aström, C. C. de Wit, M. Gäfvert, and P. Lischinsky, "Friction models and friction compensation," *European Journal of Control*, no. 4, pp. 176–195, 1998.

[18] M. Oliveira, J. Geromel, and J. Bernussou, "Extended H2 and H_∞ norm characterizations and controller parametrizations for discrete-time systems," *Int. Journal of Control*, vol. 75, no. 9, pp. 666–679, 2002.

[19] D. Robert, "Contribution to control and scheduling interaction," Ph.D. dissertation, INPG - Laboratoire d'Automatique de Grenoble (in french), january 11, 2007.